Study1_data_analysis

Tong

11/10/2021

```r
df <- read.csv("study_1_df.csv")
df_demo <- read.csv("df_data_demo2.csv")
df_demo_before <- read.csv("df_data_demo.csv")
```

Descriptive statistics about participants: what are participants like and what do they say?

See below, the comments that participants gave:

```r
unique(df_demo$comments)
```

```
##  [1] ""
##  [2] "Ingenieurswissenschaften fehlen in der Aufzählung der Studiengänge."
##  [3] "zwischendurch nach etwa drei Fragen hatte ich einen Fehler, wo die Seite auf einmal nicht mehr
##  [4] "Ggf. wäre es sinnvoll noch eine Abfrage je Fragerunde zu machen, wie sicher man sich ist. Bei m
##  [5] "nein"
##  [6] "Es war eine sehr angenehme Umfrage."
##  [7] "Die Studie war sehr übersichtlich und verständlich gestaltet."
##  [8] "Sehr spannende Studie, die zum Nachdenken anregt!"
##  [9] "Nette interaktive Studie, hat Spaß gemacht und etwas zum nachdenken angeregt. "
## [10] "Es wäre sehr interessant, die tatsächlichen Wahrscheinlichkeiten zu erfahren."
## [11] "Die Studie ist sehr verständlich und gut aufgebaut/gestaltet! "
## [12] "Einige Punkte hätte ich sehr gerne gleich gewichten können. "
## [13] "die Studie hat mir sehr gefallen & es hat sogar ein wenig Spaß gemacht, sich über solche Dinge
## [14] "Mir war nicht ganz klar ob ein Ingenieur Studium unter Mathematik und Informatik fällt."
## [15] "Die Möglichkeit, dass man mehrere Ereignisse auf die gleiche Stufe (gleiche Wahrscheinlichkeit)
## [16] "Hat spaß gemacht"
## [17] "fands tatsächlich interessant (passiert selten genug) und werde ein paar der \"fragen\", an die
```

See below, the comments that all participants gave without any exclusions: - I will not exclude this in the manuscript, just for our information.

```r
unique(df_demo_before$comments)
```

```
##  [1] ""
##  [2] "Ingenieurswissenschaften fehlen in der Aufzählung der Studiengänge."
##  [3] "zwischendurch nach etwa drei Fragen hatte ich einen Fehler, wo die Seite auf einmal nicht mehr
##  [4] "Interessante Studie"
##  [5] "Ggf. wäre es sinnvoll noch eine Abfrage je Fragerunde zu machen, wie sicher man sich ist. Bei m
##  [6] "nein"
##  [7] "Es war eine sehr angenehme Umfrage."
##  [8] "Die Studie war sehr übersichtlich und verständlich gestaltet."
##  [9] "Sehr spannende Studie, die zum Nachdenken anregt!"
## [10] "Nette interaktive Studie, hat Spaß gemacht und etwas zum nachdenken angeregt. "
## [11] "Es wäre sehr interessant, die tatsächlichen Wahrscheinlichkeiten zu erfahren."
```

```
## [12] "War eine gute Studie und man war gezwungen alles ordentlich durchzulesen. jedoch wünsche ich mi
## [13] "Die Studie ist sehr verständlich und gut aufgebaut/gestaltet! "
## [14] "Einige Punkte hätte ich sehr gerne gleich gewichten können. "
## [15] "die Studie hat mir sehr gefallen & es hat sogar ein wenig Spaß gemacht, sich über solche Dinge
## [16] "Mir war nicht ganz klar ob ein Ingenieur Studium unter Mathematik und Informatik fällt."
## [17] "Waren ziemlich interessante aufgaben aber etwas schwere entscheidungen."
## [18] "Die Möglichkeit, dass man mehrere Ereignisse auf die gleiche Stufe (gleiche Wahrscheinlichkeit)
## [19] "Hat spaß gemacht"
## [20] "fands tatsächlich interessant (passiert selten genug) und werde ein paar der \"fragen\", an di
```

See below, MEAN and SD age of the participants

```
mean(df_demo$age %>% as.numeric())
```

```
## [1] 27.20339
```

```
sd(df_demo$age %>% as.numeric())
```

```
## [1] 9.061243
```

```
#ggplot(data_demo, aes(x=age)) + geom_bar()
```

See below the gender distribution of participants.

```
df_demo %>% group_by(gender) %>% count()
```

```
## # A tibble: 3 x 2
## # Groups:   gender [3]
##   gender         n
##   <chr>      <int>
## 1 female        90
## 2 male          84
## 3 non-binary     3
```

Descriptive statistics, multinomial probability distribution.

- The average rate of providing a ranking with logical errors is around 0.39 (or, 0.3893597 to be more precise).
- The average rate of providing a ranking with logical errors under the "ranking middle events only" condition is around 0.51 (or 0.5131827 to be more precise).

- The average rate of providing a ranking with logical errors under the "ranking edge events only" condition is around 0.27 (or 0.2655367 to be more precise).

/n

- For the "ties allowed" condition:

- The probability of giving ties is around 0.19 (0.1904762). /n

- Under the condition "ranking middle events only", the conditional probability of proving type 1 logically incorrect ranking is 0.41554054, the conditional probability of providing type 2 logically incorrect ranking is 0.5033784, and the conditional probability of providing type 3 logically incorrect ranking is 0.08108108.

- Under the condition "ranking edge events only", the conditional probability of proving type 1 logically incorrect ranking is 0.03012048, the conditional probability of providing type 2 logically incorrect ranking is 0.94578313, and the conditional probability of providing type 3 logically incorrect ranking is 0.02409639.

What if conditional on the rankings being wrong and not belong to the type rankings? When ranking indifferent events, the conditional probability of proving type 1 is 0.4522059. While when ranking extreme events only, the conditional probability of providing type 1 is 0.0308642.

/n

- For the "ties not allowed" condition:
- Under the condition "ranking middle events only", the conditional probability of proving type 1 logically incorrect ranking is 0.3815261, the conditional probability of providing type 2 logically incorrect ranking is 0.6184739.
- Under the condition "ranking edge events only", the conditional probability of proving type 1 logically incorrect ranking is 0.07758621, the conditional probability of providing type 2 logically incorrect ranking is 0.9224138.

```
head(df)
```

```
##     ID between_subject_condition within_subject_condition f00 duration
## 1 3135              ties_allowed                   indiff   1 35062.38
## 2 3135              ties_allowed                   indiff   2 85294.30
## 3 3135              ties_allowed                  extreme   3 27354.62
## 4 3135              ties_allowed                   indiff   4 20320.56
## 5 3135              ties_allowed                   indiff   5 12307.88
## 6 3135              ties_allowed                  extreme   6 25339.97
##   presentation_order   eveTopleft  eveTopright  eveDownleft eveDownright
## 1            A_b_B_a  indiff8_pos  indiff4_neg  indiff4_pos  indiff8_neg
## 2            a_b_B_A  indiff7_neg indiff11_neg indiff11_pos  indiff7_pos
## 3            A_B_b_a    impl1_pos    plau1_pos    plau1_neg    impl1_neg
## 4            a_B_b_A  indiff5_neg  indiff3_pos  indiff3_neg  indiff5_pos
## 5            A_b_a_B indiff10_pos  indiff9_neg indiff10_neg  indiff9_pos
## 6            a_B_A_b    impl3_neg    impl2_pos    impl3_pos    impl2_neg
##         rank_1       rank_2      rank_3       rank_4 if_there_are_errors
## 1  indiff4_neg  indiff8_neg indiff4_pos  indiff8_pos                   1
## 2  indiff7_neg indiff11_pos indiff7_pos indiff11_neg                   1
## 3    impl1_neg    plau1_pos   plau1_neg    impl1_pos                   0
## 4  indiff3_pos  indiff5_neg indiff3_neg  indiff5_pos                   1
## 5 indiff10_pos  indiff9_pos indiff9_neg indiff10_neg                   0
## 6    impl3_neg    impl2_neg   impl2_pos    impl3_pos                   0
##   error_type if_there_are_ties classify_all_ranks
## 1          0                 0             type_2
## 2          0                 0             type_2
## 3         NA                 0            logical
## 4          0                 0             type_2
## 5         NA                 0            logical
## 6         NA                 0            logical
```

```
str(df)
```

```
## 'data.frame':    2124 obs. of  18 variables:
##  $ ID                       : int  3135 3135 3135 3135 3135 3135 3135 3135 3135 3135 ...
##  $ between_subject_condition: chr  "ties_allowed" "ties_allowed" "ties_allowed" "ties_allowed" ...
##  $ within_subject_condition : chr  "indiff" "indiff" "extreme" "indiff" ...
##  $ f00                      : int  1 2 3 4 5 6 8 9 10 11 ...
##  $ duration                 : num  35062 85294 27355 20321 12308 ...
##  $ presentation_order       : chr  "A_b_B_a" "a_b_B_A" "A_B_b_a" "a_B_b_A" ...
##  $ eveTopleft               : chr  "indiff8_pos" "indiff7_neg" "impl1_pos" "indiff5_neg" ...
##  $ eveTopright              : chr  "indiff4_neg" "indiff11_neg" "plau1_pos" "indiff3_pos" ...
```

```
##  $ eveDownleft              : chr  "indiff4_pos" "indiff11_pos" "plau1_neg" "indiff3_neg" ...
##  $ eveDownright             : chr  "indiff8_neg" "indiff7_pos" "impl1_neg" "indiff5_pos" ...
##  $ rank_1                   : chr  "indiff4_neg" "indiff7_neg" "impl1_neg" "indiff3_pos" ...
##  $ rank_2                   : chr  "indiff8_neg" "indiff11_pos" "plau1_pos" "indiff5_neg" ...
##  $ rank_3                   : chr  "indiff4_pos" "indiff7_pos" "plau1_neg" "indiff3_neg" ...
##  $ rank_4                   : chr  "indiff8_pos" "indiff11_neg" "impl1_pos" "indiff5_pos" ...
##  $ if_there_are_errors      : int  1 1 0 1 0 0 1 1 1 1 ...
##  $ error_type               : int  0 0 NA 0 NA NA 0 0 1 1 ...
##  $ if_there_are_ties        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ classify_all_ranks       : chr  "type_2" "type_2" "logical" "type_2" ...
```

```r
mean(df$if_there_are_errors)
```

```
## [1] 0.3893597
```

```r
df %>%
  filter(within_subject_condition == "indiff") %>%
  summarise(mean_error = mean(if_there_are_errors))
```

```
##   mean_error
## 1  0.5131827
```

```r
df %>%
  filter(within_subject_condition == "extreme") %>%
  summarise(mean_error = mean(if_there_are_errors))
```

```
##   mean_error
## 1  0.2655367
```

```r
df_ties_allowed <- df %>% filter(between_subject_condition == "ties_allowed")

## function to calculate conditional prob conditional on already being wrong
con_prob_error_type <- function(df){

  no_of_rankings_with_a_logical_error <- df %>%
  select(ID, error_type) %>%
  drop_na() %>%
  nrow()


  no_of_type1 <- df %>%
  select(ID, error_type) %>%
  drop_na() %>%
  filter(error_type == "1") %>%
  nrow()

  no_of_type2 <- df %>%
  select(ID, error_type) %>%
  drop_na() %>%
  filter(error_type == "0") %>%
  nrow()

  no_of_type3 <- df %>%
  select(ID, error_type) %>%
  drop_na() %>%
  filter(error_type == "2") %>%
  nrow()
```

```r
  con_prob_type1 <- no_of_type1/no_of_rankings_with_a_logical_error
  con_prob_type2 <- no_of_type2/no_of_rankings_with_a_logical_error
  con_prob_type3 <- no_of_type3/no_of_rankings_with_a_logical_error

  return(c(con_prob_type1,  con_prob_type2,  con_prob_type3))
}



## apply the above two functions
con_prob_error_type(df_ties_allowed %>%
  filter(within_subject_condition == "indiff") )
```

```
## [1] 0.41554054 0.50337838 0.08108108
```

```r
con_prob_error_type( df_ties_allowed %>%
  filter(within_subject_condition == "extreme") )
```

```
## [1] 0.03012048 0.94578313 0.02409639
```

```r
df_ties_not_allowed <- df %>% filter(between_subject_condition == "ties_not_allowed")


con_prob_error_type(df_ties_not_allowed %>%
  filter(within_subject_condition == "indiff") )
```

```
## [1] 0.3815261 0.6184739 0.0000000
```

```r
con_prob_error_type(df_ties_not_allowed %>%
  filter(within_subject_condition == "extreme") )
```

```
## [1] 0.07758621 0.92241379 0.00000000
```

```r
##  calculate the prob. of providing ties.
mean(df_ties_allowed$if_there_are_ties)
```

```
## [1] 0.1904762
```

```r
##  another way to calculate con prob for the condition where ties are allowed

 df_ties_allowed %>%
  filter( within_subject_condition == "indiff" ) %>%
   select(ID, error_type) %>%
  drop_na() %>%
  filter(error_type != 2) %>%
  summarise(con_type1 = mean(error_type),
            con_type2 = 1-con_type1)
```

```
##   con_type1 con_type2
## 1 0.4522059 0.5477941
```

```r
  df_ties_allowed %>%
  filter( within_subject_condition == "extreme" ) %>%
   select(ID, error_type) %>%
  drop_na() %>%
  filter(error_type != 2) %>%
  summarise(con_type1 = mean(error_type),
```

```
            con_type2 = 1-con_type1)
```

```
##   con_type1 con_type2
## 1 0.0308642 0.9691358
```

```
with(df, table(if_there_are_errors, error_type, useNA = "ifany"))
```

```
##                      error_type
## if_there_are_errors    0    1    2 <NA>
##                    0    0    0    0 1297
##                    1  567  232   28    0
```

```
with(df, table(between_subject_condition, classify_all_ranks, useNA = "ifany"))
```

```
##                            classify_all_ranks
## between_subject_condition logical type_1 type_2 type_3
##          ties_allowed         630    128    306     28
##          ties_not_allowed     667    104    261      0
```

```
mtab <- df %>%
  group_by(within_subject_condition, between_subject_condition) %>%
  count(classify_all_ranks) %>%
  mutate(prop = n / sum(n))

mtab %>%
  pivot_wider(
    id_cols = c(between_subject_condition, within_subject_condition),
    names_from = classify_all_ranks,
    values_from = prop)
```

```
## # A tibble: 4 x 6
## # Groups:   within_subject_condition, between_subject_condition [4]
##   between_subject_condit~ within_subject_condit~ logical  type_1 type_2   type_3
##   <chr>                   <chr>                    <dbl>   <dbl>  <dbl>    <dbl>
## 1 ties_allowed            extreme                  0.696 0.00916  0.288  0.00733
## 2 ties_not_allowed        extreme                  0.775 0.0174   0.207 NA
## 3 ties_allowed            indiff                   0.458 0.225    0.273  0.0440
## 4 ties_not_allowed        indiff                   0.517 0.184    0.298 NA
```

```
df %>%
  filter(classify_all_ranks != "logical") %>%
  group_by(within_subject_condition, between_subject_condition) %>%
  count(classify_all_ranks) %>%
  mutate(prop = n / sum(n)) %>%
  pivot_wider(
    id_cols = c(between_subject_condition, within_subject_condition),
    names_from = classify_all_ranks,
    values_from = prop)
```

```
## # A tibble: 4 x 5
## # Groups:   within_subject_condition, between_subject_condition [4]
##   between_subject_condition within_subject_condition type_1 type_2  type_3
##   <chr>                     <chr>                     <dbl>  <dbl>   <dbl>
## 1 ties_allowed              extreme                  0.0301  0.946  0.0241
## 2 ties_not_allowed          extreme                  0.0776  0.922 NA
## 3 ties_allowed              indiff                   0.416   0.503  0.0811
## 4 ties_not_allowed          indiff                   0.382   0.618 NA
```

Analysis DV1: if there are logical errors or not.

```
a1 <- aov_ez("ID", "if_there_are_errors", df,  between = "between_subject_condition", within  = "within_
```

```
## Converting to factor: between_subject_condition
```

```
## Contrasts set to contr.sum for the following variables: between_subject_condition
```

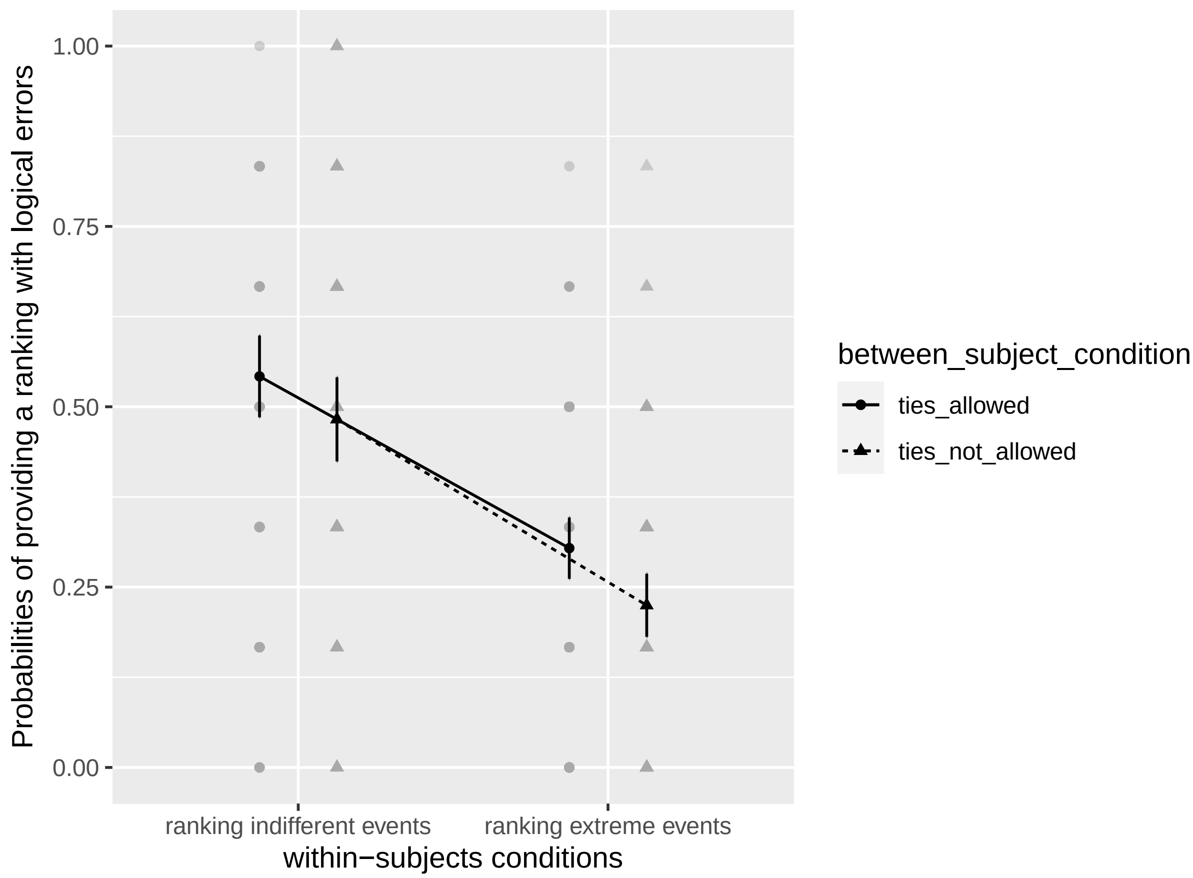
```
a1
```

```
## Anova Table (Type 3 tests)
##
## Response: if_there_are_errors
##                                              Effect     df  MSE          F
## 1                         between_subject_condition 1, 175 0.08     5.56 *
## 2                          within_subject_condition 1, 175 0.04 132.83 ***
## 3 between_subject_condition:within_subject_condition 1, 175 0.04       0.21
##     ges p.value
## 1  .020    .019
## 2  .209   <.001
## 3 <.001    .648
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

```
emmeans(a1, c("within_subject_condition", "between_subject_condition"))
```

```
##  within_subject_condition between_subject_condition emmean     SE  df lower.CL
##  indiff                   ties_allowed               0.542 0.0287 175    0.485
##  extreme                  ties_allowed               0.304 0.0216 175    0.261
##  indiff                   ties_not_allowed           0.483 0.0296 175    0.424
##  extreme                  ties_not_allowed           0.225 0.0222 175    0.181
##  upper.CL
##     0.599
##     0.347
##     0.541
##     0.269
##
## Confidence level used: 0.95
```

```
# afex plot
afex_plot(a1, "within_subject_condition", "between_subject_condition") +
  ylab(expression(paste("Probabilities of providing a ranking with logical errors"))) +
  xlab("within-subjects conditions") +
  theme(plot.margin = margin(l = 20)) +
  scale_x_discrete(labels=c("indiff" = "ranking indifferent events", "extreme" = "ranking extreme event
```




```
# people are more error-prone under condition A, where ties are allowed

ggsave("p1.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Analysis DV2: conditional probabilities of making type 1 errors giving that there are errors in the rankings.

```
## not sure if we can integrate two between-subject conditions.
 DV2_df <- df %>%
   select(ID, between_subject_condition, within_subject_condition, error_type) %>%
  drop_na() %>%
  filter(error_type != 2)


a2 <- aov_ez("ID", "error_type", DV2_df, between = "between_subject_condition", within  = "within_subje
```

```
## Converting to factor: between_subject_condition

## Warning: More than one observation per cell, aggregating the data using mean
## (i.e, fun_aggregate = mean)!

## Warning: Missing values for following ID(s):
## 2921, 2924, 2929, 2930, 2940, 2942, 2948, 2951, 2965, 2966, 2974, 2987, 2994, 2997, 2998, 3002, 3004
## Removing those cases from the analysis.

## Contrasts set to contr.sum for the following variables: between_subject_condition
```

```r
a2
```

```
## Anova Table (Type 3 tests)
##
## Response: error_type
##                                                Effect     df  MSE          F
## 1                            between_subject_condition 1, 124 0.09       0.16
## 2                             within_subject_condition 1, 124 0.08 120.70 ***
## 3 between_subject_condition:within_subject_condition 1, 124 0.08       1.66
##     ges p.value
## 1 <.001    .690
## 2  .306   <.001
## 3  .006    .200
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

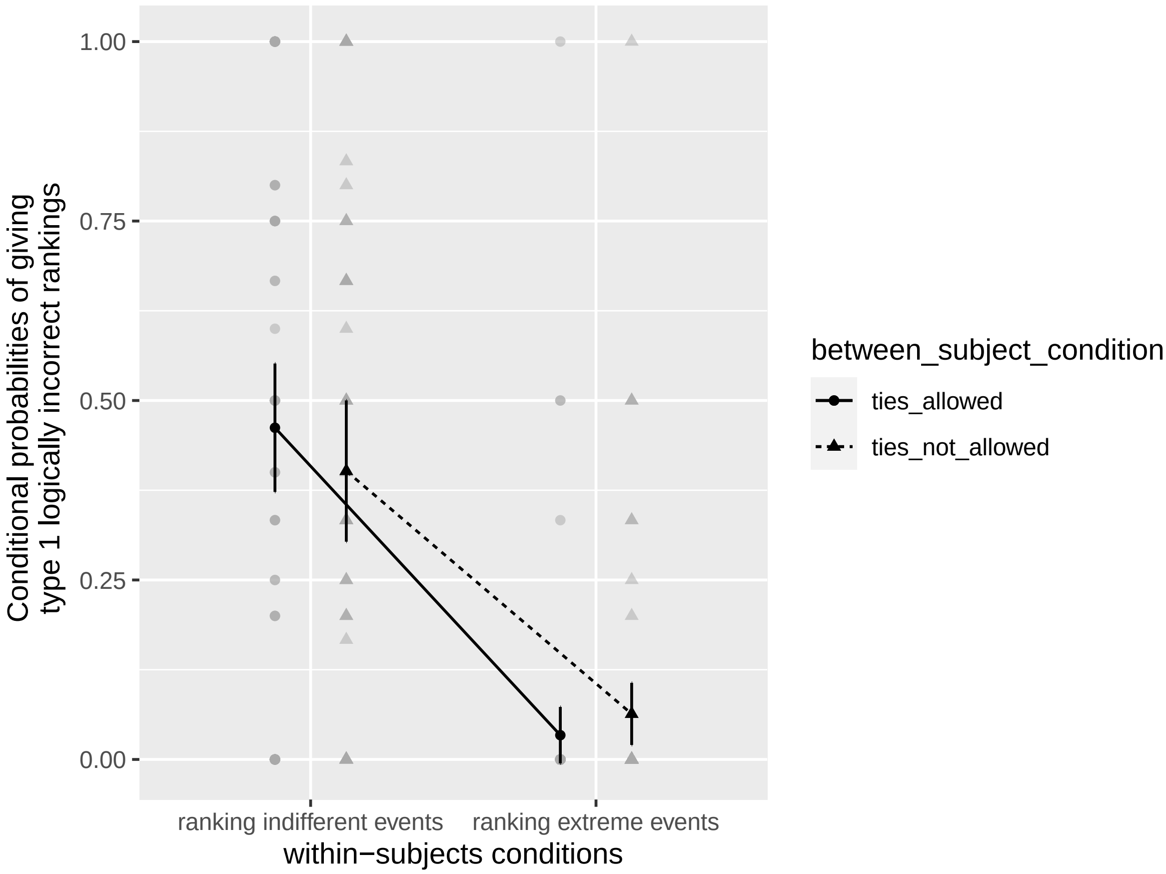
```r
emmeans(a2, c("within_subject_condition", "between_subject_condition"))
```

```
##  within_subject_condition between_subject_condition emmean     SE  df lower.CL
##  indiff                   ties_allowed              0.4621 0.0452 124  0.37264
##  extreme                  ties_allowed              0.0338 0.0199 124 -0.00551
##  indiff                   ties_not_allowed          0.4018 0.0497 124  0.30335
##  extreme                  ties_not_allowed          0.0635 0.0219 124  0.02018
##  upper.CL
##    0.5515
##    0.0731
##    0.5002
##    0.1067
##
## Confidence level used: 0.95
```

```r
# afex plot
afex_plot(a2, "within_subject_condition", "between_subject_condition") +
  ylab(expression(paste("Conditional probabilities of giving \n type 1 logically incorrect rankings")))
  xlab("within-subjects conditions") +
  theme(plot.margin = margin(l = 40)) +
  scale_x_discrete(labels=c("indiff" = "ranking indifferent events", "extreme" = "ranking extreme events
```

```
## Warning: Panel(s) show a mixed within-between-design.
## Error bars do not allow comparisons across all means.
## Suppress error bars with: error = "none"

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## metrics unknown for character 0xa

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font metrics unknown for character 0xa

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font metrics unknown for character 0xa
```




```
# first we do it separately for two datasets, namely df_ties_allowed and namely df_ties_not_allowed

# Let's start with df_ties_allowed

ggsave("conditional probability.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Analysis DV3: the probabilities of giving type 1 errors

Analysis DV4: the probabilities of giving ties. Can only analysis this DV with participants in the "ties_allowed" condition.

- For the "ties allowed" condition:
- The probability of giving ties is 0.1904762.

To do for me:

- complete the simulation for "non-tied" events with sample size ranging from 1-20, 25, 30, 35. — 50. This task is now running
- complete the simulation to investigate what is the probabilities of providing ties if we range the sample size from 1-20, 25, 30, 35. — 50.
- informative hypothesis testing (Herbert Hoijtink), allows us to test for example, |type2 − type1| > |type1 − type3|